UNCLASSIFIED

Defense Technical Information Center Compilation Part Notice

ADP012009

TITLE: Non-Stationary Subdivision for Inhomogeneous Order Differential Equations

DISTRIBUTION: Approved for public release, distribution unlimited

This paper is part of the following report:

TITLE: International Conference on Curves and Surfaces [4th], Saint-Malo, France, 1-7 July 1999. Proceedings, Volume 2. Curve and Surface Fitting

To order the complete compilation report, use: ADA399401

The component part is provided here to allow users access to individually authored sections of proceedings, annals, symposia, etc. However, the component should be considered within the context of the overall compilation report and not as a stand-alone technical report.

The following component part numbers comprise the compilation report:
ADP011967 thru ADP012009

UNCLASSIFIED

Non-Stationary Subdivision for Inhomogeneous Order Differential Equations

Joe Warren and Henrik Weimer

Abstract. This paper provides a methodology for the systematic derivation of subdivision schemes that model solutions to inhomogeneous order linear differential equations. In previous work, we showed that subdivision can be used to capture very efficiently the solutions of homogeneous order, linear differential equations. The resulting subdivision masks are stationary and can be precomputed, allowing for very simple and fast application of these schemes. In this paper, we show that this method can be extended to express solutions of systems of inhomogeneous order, linear differential equations. Even though the resulting subdivision masks may be non-stationary, the masks can again be precomputed. Thus, the resulting subdivision schemes capture very efficiently solutions of inhomogeneous order, linear partial differential equations.

§1. Subdivision for the Modeling of Shapes

Subdivision is a popular and efficient method for modeling shapes. In particular, subdivision describes a continuous shape $\boldsymbol{p}$ as the limit of a sequence p_k, $k \geq 0$ of discrete shapes,

$$\lim_{k \to \infty} p_k = \boldsymbol{p}.$$

The beauty of subdivision lies in the fact that these discrete shapes p_k are linked by a simple linear transformation S which is based on splitting and averaging,

$$p_k = S_{k-1} p_{k-1}.$$

Figure 1 shows an example of a subdivision scheme. Starting from the coarse shape p_0 on the left, application of the subdivision matrix S_0 yields the denser shape p_1. As we continue the process, the sequence of discrete shapes

Curve and Surface Fitting: Saint-Malo 1999
Albert Cohen, Christophe Rabut, and Larry L. Schumaker (eds.), pp. 411–420.

ISBN 0-8265-1357-3.

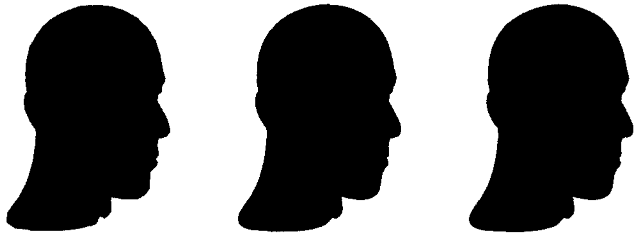

Fig. 1. Subdivision models a shape as the limit of a sequence of discrete shapes.

converges rapidly to a continuous shape $\boldsymbol{p}$ that follows the original coarsest shape p_0 and whose properties are determined by the subdivision matrix.

Subdivision's popularity for the modeling of curves is due to the algorithms by Chaikin [9], and Lane and Riesenfeld [6]. The breakthrough for the modeling of surfaces via subdivision was marked by the papers by Catmull and Clark [2] and by Doo and Sabin [3]. A popular subdivision scheme for modeling with triangular meshes has been proposed by Loop [7], which was also used for creating Figure 1.

§2. Shape Modeling through Differential Equations

Alternatively, shapes can be characterized as solutions to partial differential equations. For example, any polynomial spline $\boldsymbol{p}[\boldsymbol{x}]$ of degree m satisfies the differential equation $\boldsymbol{p}^{(m+1)}[\boldsymbol{x}] = 0$, requiring the $(m+1)$st derivative of the spline to be zero everywhere except at a fixed number of knots [1]. Other examples of shapes based on partial differential equations are the polyharmonic surfaces, including Thin Plate Splines, as well as many different classes of fluid flows.

When modeling with differential equations, we determine a continuous shape $\boldsymbol{p}$ that is a solution to a set of partial differential equations

$$\boldsymbol{D}\,\boldsymbol{p} = \boldsymbol{b}, \tag{1}$$

where $\boldsymbol{D}$ denotes a continuous differential operator and $\boldsymbol{b}$ encodes the boundary conditions for the problem. For the example of natural cubic splines, we have $\boldsymbol{D} = \frac{\partial^4}{\partial \boldsymbol{x}^4}$ and $\boldsymbol{b} = 0$ almost everywhere. If all differential operators in $\boldsymbol{D}$ are of the same, fixed order, we call the differential equation homogeneous order. Otherwise, the equation is called inhomogeneous order.

To handle such problems in a computational environment, one commonly discretizes the continuous problem. To this end, a domain grid T_k is chosen and all entities of the continuous partial differential equation (1) are discretized over this domain grid. The result is a system of linear equations

$$D_k p_k = b_k, \tag{2}$$

where p_k denotes an approximation of the continuous solution $\boldsymbol{p}$ over the grid T_k, b_k denotes a discretization of the boundary conditions, and D_k is a discrete approximation of the continuous differential operators $\boldsymbol{D}$ on the domain grid T_k.

Relying on the theory for finite elements or finite differences [11], the discrete solutions p_k can be formally guaranteed to converge to the continuous solution $\boldsymbol{p}$ of the original continuous problem (1) if the discretizations T_k are chosen carefully and the discrete representations D_k and b_k are well chosen.

At this point, the problem of finding the continuous solution $\boldsymbol{p}$ of the system of continuous partial differential equations (1) has been reduced to the problem of solving denser an denser systems of linear equations (2).

The links between mesh modeling and differential equations were previously investigated by Mallet [8], Taubin [12], and Kobbelt [5]. The method presented here is new because subdivision schemes that model solutions of inhomogeneous order differential equations are precomputed entirely, enabling very efficient modeling of shapes guided by inhomogeneous order differential equations. In particular, the actual application of the subdivision schemes does not require any computational solving whatsoever.

§3. Subdivision for Homogeneous Order Differential Equations

In our previous work [13,14] we characterized subdivision schemes for the solutions of homogeneous order linear partial differential equations. In this framework, the subdivision matrix S_{k-1} is determined as the solution to the system of linear equations

$$D_k S_{k-1} = 2^d U_{k-1} D_{k-1}, \tag{3}$$

where d is the dimension of the domain. Recall that the differencing operator D_k is the discrete approximation of the continuous differential operator $\boldsymbol{D}$ of the original, continuous problem (1) on the level k grid T_k. Further, U_{k-1} denotes a very simple linear transformation, called replication or upsampling, that carries coefficients over the grid T_{k-1} into coefficients over the next denser grid T_k. The action of U_{k-1} is very simple: Coefficients centered over knots in T_{k-1} are replicated over the same knots in the denser grid T_k while coefficients centered over the remaining knots $T_k - T_{k-1}$ are set to zero. Thus, U_{k-1} is a matrix whose rows are either zero or a standard unit vector, and U_{k-1} can be constructed easily and efficiently.

We visualize the meaning of equation (3) in Figure 2: The subdivision matrix is determined so that a certain commutativity relationship holds between subdivision, upsampling and differencing. Differencing coefficients on the coarse grid and upsampling those differences to the finer grid by inserting zero for all new grid points ($U_{k-1}D_{k-1}$, the right hand side of equation (3)) should yield the same result as subdividing the coefficients using the subdivision scheme and then differencing on the finer grid ($D_k S_{k-1}$, the left hand side of equation (3)).

$$\begin{tikzcd}
p_{k-1} \arrow[r, "\text{difference}"] \arrow[d, "\text{subdivide}"'] & E_{k-1}\, p_{k-1} \arrow[d, "\text{upsample}"] \\
S_{k-1}\, p_{k-1} \arrow[r, "\text{difference}"] & E_k\, S_{k-1}\, p_{k-1} = U_k\, E_{k-1}\, p_{k-1}
\end{tikzcd}$$

Fig. 2. The subdivision scheme is determined such that this commutativity relationship holds.

The subdivision matrices S_k are the only unknowns of equation (3), and we can use linear algebra to systematically solve for these subdivision matrices. In our previous work [13] we showed that solutions produced by these subdivision matrices are related to solutions produced by an interpolating finite element solver using a simple, fixed change of basis. Therefore, if the finite element solver converges, then the subdivision solution is also well defined.

As a side note, in our previous work [13] we establish that the right-hand side of the system, as solved by the subdivision scheme from relation (3), is $D_k p_k = U_{k-1} U_{k-2} \ldots U_0 D_0 p_0$ where $p_k = S_{k-1} p_{k-1}$ and p_0 is a user-given set of initial control coefficients. In other words, the subdivision scheme leads to a specific combination of the integer shifts of the Green function of the differential operator.

Further, computation of the subdivision matrix S_{k-1} based on relation (3) requires the inversion of the differencing operator D_k. Consequently, the computational work required for finding the subdivision matrix is at least the same as inverting the finite difference system. However, later we will see that the subdivision matrices can be precomputed. Thus, in contrast to a conventional finite difference solver, new shapes can be generated extremely efficiently.

As an example, we briefly derive subdivision schemes for piecewise polynomial splines. Recall from deBoor [1] that the piecewise polynomial spline $\boldsymbol{p}[\boldsymbol{x}]$ of degree m satisfies the differential equation $\boldsymbol{D}[\boldsymbol{x}]^{m+1}\boldsymbol{p}[\boldsymbol{x}] = 0$ where $\boldsymbol{D}[\boldsymbol{x}]$ denotes the first derivative in the variable $\boldsymbol{x}$.

We employ generating functions [4] for concise and convenient encoding of discrete coefficient sequences. To this end, we choose the domain grids for our analysis as the dilates $\frac{1}{2^k}\mathbb{Z}$ of the integer grid $\mathbb{Z}$. A generating function $p_k[x]$ is a power series that associates the ith coefficient of the discrete shape p_k as the coefficient of x^i. For example, the coefficient sequence $\{1, 2, 3, 4, 5\}$ is represented by $1 + 2x + 3x^2 + 4x^3 + 5x^4$.

Recall the definition of the first derivative operator,

$$\boldsymbol{D}[\boldsymbol{x}]\boldsymbol{p}[\boldsymbol{x}] = \lim_{\boldsymbol{t} \to 0} \frac{\boldsymbol{p}[\boldsymbol{x}] - \boldsymbol{p}[\boldsymbol{x} + \boldsymbol{t}]}{\boldsymbol{t}}.$$

Substituting $\boldsymbol{t} = \frac{1}{2^k}$ yields

$$\boldsymbol{D}[\boldsymbol{x}]\boldsymbol{p}[\boldsymbol{x}] = \lim_{k\to\infty} \frac{\boldsymbol{p}[\boldsymbol{x}] - \boldsymbol{p}\left[\boldsymbol{x} + \frac{1}{2^k}\right]}{\frac{1}{2^k}}.$$

Thus, for $\boldsymbol{x} \in \frac{1}{2^k}\mathbb{Z}$, the approximation of the first derivative is given by the difference between two adjacent discretizations, normalized by the grid spacing. In terms of generating functions, this differencing operation is represented by the Laurent polynomial

$$D_k[x] = 2^k \frac{1-x}{x^{1/2}}.$$

Higher order derivatives and differences are obtained by repeated application of the respective continuous or discrete operator.

In terms of generating functions, the action of the upsampling operator U_k can be captured very concisely: The expression $p\left[x^2\right]$ represents the upsampled coefficient sequence of $p[x]$ as a generating function. Thus, in our example of polynomial splines, the generating function $s_k[x]$ for the subdivision scheme satisfies

$$D_k[x]^{m+1} s_k[x] = 2D_{k-1}\left[x^2\right]^{m+1},$$

which can be simplified to

$$s_k[x] = 2\left(\frac{D_{k-1}[x^2]}{D_k[x]}\right)^{m+1}.$$

Fortunately,

$$\frac{D_{k-1}[x^2]}{D_k[x]} = \frac{1}{2}\,\frac{1+x}{x^{1/2}},$$

i.e. the generating functions for the differencing operations on the level $k-1$ and level k grids divide out yielding a simple expression independent of k. As a result, the subdivision mask for the degree m polynomial splines are exactly the coefficients of

$$s[x] = 2\left(\frac{1+x}{2x^{1/2}}\right)^{m+1}.$$

Remarkably, these are precisely the known subdivision schemes for piecewise linear functions ($m = 1$), Chaikin's algorithm [9] ($m = 2$) and the Lane/Riesenfeld algorithm [6] ($m = 3$).

Previously we applied this strategy to derive subdivision schemes modeling solutions of homogeneous order linear differential equations yielding local, stationary subdivision masks [13,14]. In this paper, we show that largely the same strategy can be used to determine subdivision schemes for inhomogeneous order linear partial differential equations. As we will see, in this case the actual subdivision masks may depend on the particular level of subdivision, i.e. are non-stationary. However, the masks can still be precomputed as a closed form algebraic expression in the level of subdivision, which can then be evaluated very efficiently during the actual application of the scheme.

§4. Subdivision for Inhomogeneous Order Differential Equations

In this section we extend our systematic construction of subdivision schemes to handle inhomogeneous order linear partial differential equations. We consider the simple yet interesting problem of splines in tension [10]. The continuous spline in tension $\boldsymbol{p}[\boldsymbol{x}]$ for tension parameter α satisfies the differential equation

$$\left(\boldsymbol{D}[\boldsymbol{x}]^4 - \alpha^2\ \boldsymbol{D}[\boldsymbol{x}]^2\right) \boldsymbol{p}[\boldsymbol{x}] = 0, \tag{4}$$

where $\boldsymbol{D}[\boldsymbol{x}]$ again represents the continuous first derivative operator with respect to the variable $\boldsymbol{x}$. Note that equation (4) incorporates both second and fourth derivatives of $\boldsymbol{p}[\boldsymbol{x}]$, i.e. is inhomogeneous order.

Following the same strategy as in the derivations for polynomial splines, we use generating functions to encode the discrete approximation p_k of the spline in tension on grid T_k as well as for the representation of the differencing operation $D_k[x] = 2^k \frac{1-x}{x^{1/2}}$. Next, we apply equation (3) to characterize the subdivision scheme $s_k[x]$ as the solution to

$$\left(D_k[x]^4 - \alpha^2 D_k[x]^2\right) s_{k-1}[x] = 2\left(D_{k-1}\left[x^2\right]^4 - \alpha^2 D_{k-1}\left[x^2\right]^2\right), \tag{5}$$

which can be simplified to

$$s_{k-1}[x] = \frac{2\left(D_{k-1}[x^2]^4 - \alpha^2 D_{k-1}[x^2]^2\right)}{D_k[x]^4 - \alpha^2 D_k[x]^2}. \tag{6}$$

However, at this point we note that there is no simple closed-form expression for the quotient $\frac{D_{k-1}[x^2]^4 - \alpha^2 D_{k-1}[x^2]^2}{D_k[x]^4 - \alpha^2 D_k[x]^2}$ (unless $\alpha = 0$). In other words, there is no finitely-supported subdivision scheme $s_{k-1}[x]$ for splines in tension. Moreover, the coefficients of the Laurent series expansion of the quotient $s_{k-1}[x]$ depend on the level of subdivision k, i.e. the subdivision scheme has to be non-stationary.

Fortunately, due to the structure of equation (3) the coefficients of this expansion decrease very rapidly away from the origin. Thus, we can approximate the infinite Laurent expansion of the subdivision mask well by a locally supported scheme. To this end, we construct the generating function $s_{k-1}[x]$ of desired support symbolically with the actual coefficients s^i_{k-1} as unknowns,

$$s_{k-1}[x] = \sum_{i=-n}^{n} s^i_{k-1} x^i$$

for a user-defined support n. We then construct a generating function for the residual of equation (5),

$$\begin{aligned} r_k[x] &= \left(D_k[x]^4 - \alpha^2 D_k[x]^2\right) s_{k-1}[x] - 2\left(D_{k-1}\left[x^2\right]^4 - \alpha^2 D_{k-1}\left[x^2\right]^2\right) \\ &= \sum_i r^i_k x^i. \end{aligned} \tag{7}$$

Using linear algebra, we can now solve for the unknowns s^i_{k-1} of (7) symbolically by minimizing the least squares residual of the coefficients r^i_k. The motivation behind our strategy is to construct a best solution for the characteristic equation (3) of given support. The results of this process are actual, symbolic coefficients for the local subdivision scheme $s_{k-1}[x]$, depending on the tension parameter α as well as on the level k. As an example, the approximation to (6) with the same support as the Lane-Riesenfeld algorithm ($n = 2$) has

$$\frac{2^{2k}\left(693\,2^{1+10k}+891\,4^{1+4k}\,\alpha^2+3525\,64^k\,\alpha^4+399\,4^{1+2k}\,\alpha^6+333\,4^k\,\alpha^8+26\,\alpha^{10}\right)}{8\left(693\,2^{1+12k}+891\,4^{1+5k}\,\alpha^2+3861\,256^k\,\alpha^4+273\,2^{3+6k}\,\alpha^6+675\,16^k\,\alpha^8+27\,4^{1+k}\,\alpha^{10}+7\,\alpha^{12}\right)}$$

as the coefficient for $x^{\pm 2}$,

$$\frac{\left(4^{1+k}+\alpha^2\right)\left(693\,1024^k+171\,4^k\,\alpha^8+2\,\alpha^2\left(891\,256^k+219\,4^{1+3k}\,\alpha^2+399\,16^k\,\alpha^4+7\,\alpha^8\right)\right)}{4\left(693\,2^{1+12k}+891\,4^{1+5k}\,\alpha^2+3861\,256^k\,\alpha^4+273\,2^{3+6k}\,\alpha^6+675\,16^k\,\alpha^8+27\,4^{1+k}\,\alpha^{10}+7\,\alpha^{12}\right)}$$

as the coefficient for $x^{\pm 1}$, and finally

$$\frac{2079\,4^{1+6k}+6039\,4^{1+5k}\,\alpha^2+1755\,16^{1+2k}\,\alpha^4+8265\,2^{1+6k}\,\alpha^6+5235\,16^k\,\alpha^8+213\,4^{1+k}\,\alpha^{10}+56\,\alpha^{12}}{8\left(693\,2^{1+12k}+891\,4^{1+5k}\,\alpha^2+3861\,256^k\,\alpha^4+273\,2^{3+6k}\,\alpha^6+675\,16^k\,\alpha^8+27\,4^{1+k}\,\alpha^{10}+7\,\alpha^{12}\right)}$$

as the coefficient associated with x^0. Note that for $\alpha = 0$ these coefficients exactly reduce to the subdivision scheme for natural cubic splines based on the Lane-Riesenfeld algorithm.

During an actual application of the subdivision scheme, the user-defined tension parameter α and the current level of subdivision k are substituted into the symbolic solution $s_{k-1}[x]$, yielding a simple generating function in only the variable x. The coefficients of this generating function encode the subdivision masks for the spline in tension for the given tension parameter α at the current level k. Again, application of this subdivision scheme is very efficient. For example, given $\alpha = 0$, the above expression simplifies to the generating function for natural cubic spline subdivision, independent of k.

$k=1:$	0.11063	0.55261	0.88274	0.55261	0.11063
$k=2:$	0.12345	0.52441	0.80178	0.52441	0.12345
$k=3:$	0.12489	0.50733	0.76487	0.50733	0.12489
$k=4:$	0.12499	0.50192	0.75386	0.50192	0.12499
$k=5:$	0.125	0.50049	0.75097	0.50049	0.125
$k=6:$	0.125	0.50012	0.75024	0.50012	0.125
$k=7:$	0.125	0.50003	0.75006	0.50003	0.125
$k=8:$	0.125	0.50001	0.75002	0.50001	0.125
$k=9:$	0.125	0.5	0.75	0.5	0.125
$k=10:$	0.125	0.5	0.75	0.5	0.125
$k=11:$	0.125	0.5	0.75	0.5	0.125

Fig. 3. Subdivision masks for $\alpha = 1$, $k = 0$.

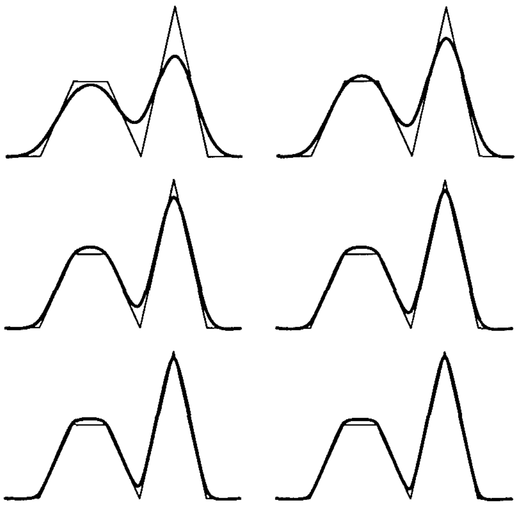

Fig. 4. Splines in tension for varying α.

Figure 3 shows the actual coefficients of a locally supported generating function ($n = 2$) for $\alpha = 1$ and $k = 1, \ldots, 11$. Coefficients were rounded to five significant digits. Note that the coefficient sequence rapidly converges to the subdivision scheme for natural cubic splines. Indeed, after a few rounds of subdivision, a spline in tension behaves like a natural cubic spline over a denser initial grid with its initial control coefficients determined by the first few rounds of subdivision.

Figure 4 depicts application of four rounds of the local subdivision scheme (support $n = 4$) for α ranging from 0 to 5. The initial control polygon is shown as a thin line, while the subdivided curve is shown in solid. Note that as α is increased, the curve follows the control polygon more closely. In the limit, $\alpha \to \infty$, the curve is actually the piecewise linear interpolant of the initial control points.

Figure 5 shows the least squares residuals $\sum_i (r_k^i)^2$ of approximations of different sizes for $\alpha = 1$ and $k = 0$ (the residual is largest for $k = 0$) on a logarithmic scale. Note that for the approximation of size $n = 4$ the residual is already very small.

At a higher level, we follow these steps in the derivation of non-stationary subdivision schemes for inhomogeneous order linear partial differential equations:

Starting from the continuous, inhomogeneous order, linear partial differential equations we discretize the continuous differential operators to yield appropriate differencing operators over the respective subdivision grids T_k. We then characterize the subdivision scheme s_{k-1} as the only unknown of equation (3) using these differencing operators as well as simple replication/upsampling. Next, we construct a representation of the subdivision scheme s_{k-1} in terms

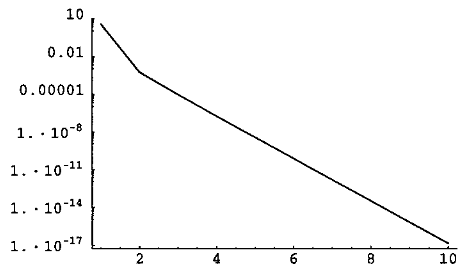

Fig. 5. Residuals of local approximations with varying support.

of unknowns and symbolically build a residual expression representing the difference between left-hand side and right-hand side of equation (3). Finally, we use linear algebra to solve symbolically for the unknowns of the subdivision scheme s_{k-1}, which may depend on the level of subdivision and possible parameters to the original partial differential equations. As a result, application of the subdivision scheme only involves instantiation of these constants, yielding a locally supported, approximating subdivision scheme for solutions of the original inhomogeneous order partial differential equations.

§5. Summary and Conclusion

In this paper, we showed that subdivision can be used to model solutions of inhomogeneous order differential equations. Using the characterization of the subdivision scheme based on the commutativity relationship (3), we can systematically solve for these schemes. Even though the exact subdivision schemes may be globally supported, locally supported schemes approximate the solution well enough for practical purposes. Non-stationary schemes can be handled using the same methodology by allowing the locally supported subdivision masks to change between levels. Because these subdivision schemes can be precomputed, the modeling of solutions of inhomogeneous order linear partial differential equations can be handled very efficiently.

The proposed method for modeling solutions to inhomogeneous order linear differential equations is quite general and promises to be useful in a variety of applications. First of all, approximations based on local subdivision schemes are often sufficient for modeling applications. Indeed, the approximate solutions are qualitatively indistinguishable from the exact solution. Second, if the accuracy of the subdivision solution is not satisfactory, the subdivision scheme can be used to produce very good initial estimates for more traditional solution methods. Third, the results of traditional solution methods often need to be refined locally for visualization and analysis. A local subdivision scheme can be used to refine solutions to any desired accuracy and provide better accuracy than traditional polynomial fits.

Acknowledgments. This work was supported in part under NSF grant number CCR-9732344.

References

1. de Boor, C., *A Practical Guide to Splines.* Springer, 1978.
2. Catmull, E. and J. Clark, Recursively generated B-spline surfaces on arbitrary topological meshes, Computer Aided Design **10** (1978), 350–355.
3. Doo, D. and M. Sabin: Behavior of recursive subdivision surfaces near extraordinary points, Computer Aided Design **10** (1978), 356–360.
4. Dyn, N., Subdivision schemes in computer aided geometric design, in *Advances in Numerical Analysis II,* W. Light (ed.), Oxford University Press, 1992, 36–104.
5. Kobbelt, L., Fairing by finite difference methods, in *Mathematical Methods for Curves and Surfaces II,* M. Daehlen, T. Lyche, and L. L. Schumaker (eds.), Vanderbilt University Press, 1998.
6. Lane, J. and R. Riesenfeld, A theoretical development for the computer generation and display of piecewise polynomial surfaces, IEEE Transactions on Pattern Analysis and Machine Intelligence **2** (1980), 35–46.
7. Loop, C. T., *Smooth Subdivision Surfaces Based on Triangles,* Master's Thesis, Department of Mathematics, University of Utah, 1987.
8. Mallet, J.-L., Discrete smooth interpolation, Transactions on Graphics **4** No. 2 (1985), 74–123.
9. Riesenfeld, R., On Chaikin's algorithm, Computer Graphics and Image Processing **4** (1975), 304–310.
10. Schweikert, D. G., An interpolating curve using a spline in tension, J. Math. and Physics **45** (1966), 312–317.
11. Shaidurov, V. V., *Multigrid Methods for Finite Elements,* Kluwer Academic Publishers, 1995.
12. Taubin, G., A signal processing approach to fair surface design, Proceedings of SIGGRAPH 95, Computer Graphics Proceedings, Annual Conference Series, Addison Wesley (1995), 351–358.
13. Warren, J. and H. Weimer, Subdivision schemes for variational problems, preprint available online under `http://www.cs.rice.edu/~henrik/`.
14. Weimer, H. and J. Warren, Subdivision schemes for fluid flow, Proceedings of SIGGRAPH 99, Computer Graphics Proceedings, Annual Conference Series, Addison Wesley (1995), 111–120.

Joe Warren and Henrik Weimer
Rice University, Department of Computer Science
P.O. Box 1892
Houston, TX 77251-1892, USA
`{jwarren,henrik}@rice.edu`